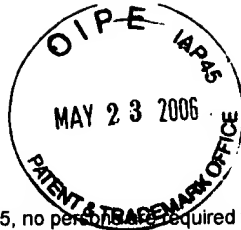


Doc Code: AP.PRE.REQ



PTO/SB/33 (07-05)

Approved for use through xx/xx/200x. OMB 0651-00xx  
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no person shall be required to respond to a collection of information unless it displays a valid OMB control number.

PRE-APPEAL BRIEF REQUEST FOR REVIEW		Docket Number (Optional)
		550-192
Application Number		Filed
09/731,060		December 7, 2000
First Named Inventor		NEVILL
Art Unit		Examiner
2194		Zhen, Li B.
<p>Applicant requests review of the final rejection in the above-identified application. No amendments are being filed with this request.</p> <p>This request is being filed with a notice of appeal.</p> <p>The review is requested for the reason(s) stated on the attached sheet(s). Note: No more than five (5) pages may be provided.</p> <p>I am the</p> <p><input type="checkbox"/> Applicant/Inventor</p> <p><input type="checkbox"/> Assignee of record of the entire interest. See 37 C.F.R. § 3.71. Statement under 37 C.F.R. § 3.73(b) is enclosed. (Form PTO/SB/96)</p> <p><input checked="" type="checkbox"/> Attorney or agent of record <u>33,149</u> (Reg. No.)</p> <p><input type="checkbox"/> Attorney or agent acting under 37CFR 1.34. Registration number if acting under 37 C.F.R. § 1.34 _____</p> <p>NOTE: Signatures of all the inventors or assignees of record of the entire interest or their representative(s) are required. Submit multiple forms if more than one signature is required, see below.*</p> <p><input checked="" type="checkbox"/> *Total of 1 form/s are submitted.</p>		

  
Signature

John R. Lastova

Typed or printed name

703-816-4025

Requester's telephone number

May 23, 2006

Date

This collection of information is required by 35 U.S.C. 132. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.11, 1.14 and 41.6. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Mail Stop AF, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

If you need assistance in completing the form, call 1-800-PTO-9199 and selection option 2.

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

In re Patent Application of

NEVILL et al.

Appl. No. 09/731,060

Filed: December 7, 2000



Atty. Ref.: 550-192; Confirmation No. 1332

TC/A.U. 2194

Examiner: Zhen, Li B.

For: SCHEDULING CONTROL WITHIN A SYSTEM HAVING MIXED HARDWARE AND  
SOFTWARE BASED INSTRUCTION EXECUTION

\* \* \* \* \*

May 23, 2006

Mail Stop AF  
Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

**STATEMENT OF ARGUMENTS IN SUPPORT OF  
PRE-APPEAL BRIEF REQUEST FOR REVIEW**

In this fourth office action, the claims stand rejected under 35 U.S.C. §103 as being unpatentable based on the combination of U.S. Patent 5,937,193 to Evoy et al described in the background of this application and an IDS article submitted by Applicants entitled, "*PicoJava: A Direct Execution Engine for Java Bytecode*," by McGhan et al. This rejection is improper.

A processing system provides both hardware instruction translation and software instruction interpretation mechanisms for supporting high level program instructions, e.g., Java bytecodes. The program instructions are all supplied to the hardware translation unit (see 64 in Fig. 9) which forwards those instructions it does not directly support to the software interpretation mechanism (see step 84 in Fig. 10). By routing all program instructions through the hardware translation unit, the hardware translation unit can monitor when it is appropriate and safe to trigger a scheduling operation for controlling multi-tasking or multi-threaded operations. Multi-tasking operating systems require processing resources to be shared between

several different programs that may be simultaneously active, and multi-threaded computer programs similarly require processing resources to be shared between different active threads.

The inventors recognized drawbacks with simplistic counter or timer scheduling approaches. A simple timer-based scheduling approach may suffer from the disadvantage that scheduling operations may be inappropriately triggered at points part-way through the software interpretation of a complex program instruction in a manner that could cause a loss of data integrity should an inappropriate context switch occur. A simple counter-based scheduling requires the extra overhead of exchanging counter values between hardware-executed program instructions and software-executed program instructions.

As recited in the independent claims, the hardware based execution unit includes scheduling support logic to generate “a scheduling signal for triggering a scheduling operation to be performed between program instructions for managing scheduling between threads or tasks irrespective of whether a preceding program instruction was executed by said hardware based execution unit or said software based execution unit.” In non-limiting examples in the specification, scheduling operations may be triggered based upon a count of executed program instructions (see, e.g., counter 70 in Fig. 9) or by using a timer-based scheduling approach (see, e.g., timer 96 in Fig. 12), with the timer signal being qualified by a signal indicating an appropriate point within the cycle of execution of program instructions.

Evoy discloses a computer system with a translator to translate platform-independent instructions into corresponding native instructions for execution by the processor. In addition to a software-based interpreter, one or more look-up tables are provided to map platform-independent instructions to native instructions. The Examiner concedes that Evoy only discloses scheduling in a general sense in that an address counter is incremented after the execution of an instruction, which has the effect of selecting the next bytecode. But as the Examiner admits,

incrementing an address counter to select the next bytecode for translation is not scheduling between threads or tasks. Indeed, Evoy suffers from the very problem described above where scheduling operations may be triggered part way through interpretation of a complex non-native program instruction.

Although the Examiner turns to McGhan to remedy this deficiency in Evoy, McGhan also fails to disclose the claimed multitasking or multithreading scheduling. McGhan's picoJava microprocessor core, like Evoy's, handles simple instructions within hardware (page 25) and complex instructions in software (page 26). But the issue to be decided in this pre-appeal hinges on a single sentence in McGhan on page 30 relied on by the Examiner: "The picoJava core also provides specialized hardware runtime support to accelerate other vital functions of the Java virtual machine, like thread management and object handling." No more is said about thread management.

Understanding the weakness of this text, the examiner speculates that because the picoJava core that directs complicated instructions to be handled by software, "the thread management of scheduling is performed irrespective of whether a preceding program instruction was executed [in hardware or software]." This speculation is clear error. McGhan does not teach the claimed feature of triggering a scheduling operation to be performed between program instructions for "managing scheduling between threads or tasks *irrespective of whether a preceding program instruction was executed by said hardware based execution unit or said software based execution unit.*" Indeed, McGhan's scheduling could very much **depend** on whether the preceding program instruction was executed by the hardware-based execution unit or by the software-based execution unit.

In fact, there is no teaching in McGhan that the picoJava core handles "thread management" in any significantly different way from known Java virtual machines. McGhan's

statement, "the picoJava core also provides specialized hardware runtime support to accelerate [functions] like thread management," suggests that the physical implementation of the picoJava core is arranged to expedite thread management processes, but in no way acknowledges the potential problems associated with scheduling in a system which executes some instructions in hardware and some instructions in software. The failure to recognize the problem being solved by the claimed invention is a telling factor pointing to a second clear error—insufficient “teaching, suggestion, or motivation to combine the references.” *In re Rouffet*, 149 F.3d 1350, 1355 (Fed. Cir. 1998).

A third clear error is the standard of obviousness employed by the examiner. If the standard for obviousness was one of speculation, one might just as easily speculate that since McGhan describes the slow excursions into software routines for handling complex instruction as "happily infrequent," (page 26), new scheduling triggers are simply forbidden when these infrequent events take place. That would circumvent the need for thread management to cope with the fact that some instructions are handled by software. But ever since the famous Supreme Court decision in *Graham v. John Deere*, the obviousness standard has been based on objective evidence. The burden to produce that evidence is squarely on the USPTO. *In re Piasecki*, 223 USPQ 785, 788 (Fed. Cir. 1984). McGhan's lacks objective evidence teaching the claimed scheduling support logic. There is no teaching of a scheduling signal that triggers a scheduling operation for “managing scheduling between threads or tasks *irrespective of whether a preceding program instruction was executed by said hardware based execution unit or said software based execution unit.*”

Given the clear errors in the rejection , the rejections should be withdrawn and the application passed to allowance.

Nevill et al  
Appl. No. 09/731,060  
May 23, 2006

Respectfully submitted,

**NIXON & VANDERHYE P.C.**

By: \_\_\_\_\_



John R. Lastova  
Reg. No. 33,149

JRL:maa  
901 North Glebe Road, 11th Floor  
Arlington, VA 22203-1808  
Telephone: (703) 816-4000  
Facsimile: (703) 816-4100